HUB-VM102

Programming Manual

Document version 1.3 | Publication date:
01. April 2025

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Table of Contents

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Legal information

Safety information

This documentation contains information that you must observe for your personal safety and to prevent material damage. Read the safety information carefully and always keep this documentation within easy reach.

The safety information is presented in descending order of hazard level as follows:

DANGER
Indicates an immediate hazard to humans. Failure to comply will lead to irreversible injuries or death.

WARNING
Indicates an identifiable hazard to humans. Failure to comply may lead to irreversible injuries or death.

CAUTION
Indicates an identifiable hazard to humans or potential material damage. Failure to comply may lead to reversible injuries or material damage.

ATTENTION
Indicates potential material damage. Failure to comply may lead to material damage.

NOTE
Notes give you tips, recommendations and useful information on specific actions and issues.

TIP
A tip gives you tips, tricks and recommendations from in.hub that have proven to be helpful in handling the products.

Qualified personnel

The product associated with this documentation may only be handled by personnel qualified for the respective task. The device may only be installed, commissioned and operated in compliance with the associated documentation and the safety information contained therein.

Based on their training and experience, qualified personnel are able to recognize risks and avoid potential hazards when handling these products.

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Knowledge of PCs, operating systems and web applications is a prerequisite. General knowledge in the field of automation technology is recommended.

Intended use

in.hub products may only be used for the applications specified in the corresponding technical documentation.

If third-party products and components are used, they must be recommended or approved by in.hub.

Proper storage, set-up, assembly, installation, commissioning, operation and maintenance are essential for the correct and safe operation of the products.

The permissible ambient conditions must be complied with. Instructions in the associated documentation must be followed.

Brands

All designations marked with the “®” symbol are registered trademarks. The other designations in this document may be trademarks whose use by third parties for their own purposes may infringe the rights of the owner.

Disclaimer

in.hub accepts no liability for product malfunctions resulting from improper handling, mechanical damage, incorrect application and improper use.

The contents of this document have been checked for conformity with the product described. However, deviations cannot be ruled out, so that we cannot guarantee complete conformity. The information in this publication is regularly reviewed. Necessary corrections are included in subsequent editions.

VORLÄUFIGE VERSION / PRELIMINARY VERSION

1. General instructions for use

This programming manual provides support if you cannot use the system software SIINEOS to set up the HUB-VM102 because:

- you do not have a master gateway
- SIINEOS does not offer the function you need

1.1. Other applicable documents

This programming manual is only valid in conjunction with the Operating Instructions for the HUB-VM102. Please read the operating instructions carefully and keep them to hand if you are programming the hardware yourself.

You can download the current operating instructions from the download portal: https://download.inhub.de/vm102/

VORLÄUFIGE VERSION / PRELIMINARY VERSION

2. Configuration of the interfaces

If you are using a HUB-VM102 and transferring the vibration sensor data to a in.hub master gateway, configure the interfaces in the I/O management of SIINEOS, which is installed on the master gateway. In this case, you do not need this programming manual, but use the user manual from SIINEOS instead. You can find it in the download portal at https://download.inhub.de/siineos/.

If you use your own or third-party devices and want to connect the HUB-VM102 to them, the internal configuration is carried out via a parameter set that can be modified via the HUB-VM102's interfaces. The parameter configuration is retained in the device even after the power supply is disconnected. You can find out how to do this in the following chapters. Please note, however, that the saving of parameters must be initiated by you and does not take place automatically.

2.1. Configuration of the Modbus parameters

Configuration and data exchange between HUB-VM102 and the gateway takes place via Modbus. The backplane bus allows communication via Modbus RTU, while the Ethernet interface supports the Modbus TCP protocol. Data can also be exchanged via the MQTT protocol.

Configuration is carried out using parameter numbers between 0 and 127. The parameters can be read out and changed using the following Modbus function codes:

- Read holding registers (0x03)
- Write multiple registers (0x10)

The configuration and control parameters are 32 bits wide. The Modbus protocol is therefore used to access the upper and lower 16 bits via consecutive Modbus addresses:

Modbus address	Parameter
0x00 (bits 0–15), 0x01 (bits 16–31)	P0
0x02, 0x03	P1
0x04, 0x05	P2
...	...
0xFE, 0xFF	P127

Changed parameters can be saved in the internal EEPROM. This means that they are available even after an interruption in the power supply.

1. Switch the device to standby mode before saving the parameters:
   P100 = 0 (device off)
2. The parameters can then be permanently saved in the EEPROM:
   P102 = 1 (store parameter)

VORLÄUFIGE VERSION / PRELIMINARY VERSION

The Modbus interface can be configured via parameters 96 and 97:

Modbus ID	P96	1–254 If the parameter is outside the permissible range, the default value (1) is used
Modbus baud rate	P97	9600–115200–1000000 If the parameter is outside the permissible range, the default value (115200) is used

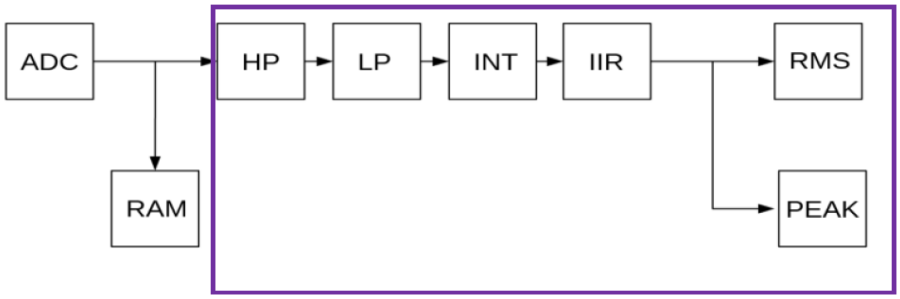
There are also parameters 91–93 for Modbus TCP:

IP address	P91	192.168.1.200 (default)
Net mask	P92	255.255.255.0 (default)
Gateway address	P93	192.168.1.1 (default)

NOTE
Parameters 91–93, 96 and 97 only become effective after a reset. Therefore, save the parameters in the EEPROM after changing them.

2.2. Signal processing

Depending on the parameter configuration, the signal processing of each channel is carried out according to the following scheme:



IEPE channel 1

Signal	Explanation	Configuration parameter	Threshold
HP	1st-order high-pass filter to eliminate the DC offset	P13, P23	100–1000–10000 mHz
INT	Integrator for determining the vibration velocity	P14, P24	100–1000–10000 mHz
LP	1st-order low-pass filter	P15, P25	100–5000–10000 Hz

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Signal	Explanation	Configuration parameter	Threshold
IIR	Digital IIR filter	See tables in chapter Configuration of an IIR filter [11]	
RMS	Root mean square value	P16, P26	100–1000–10000 mHz
PEAK	Peak value	P31, P41 P32, P42	100–1000–10000 µs 10–100–1000 ms

The current RMS value can be read out via parameters 1 and 2. The current peak value is displayed via parameters 3 and 4.

Parameter	Explanation	Resolution
P1	RMS value of channel 1	µV
P2	RMS value of channel 2	µV
P3	Peak value of channel 1	µV
P4	Peak value of channel 2	µV
P5	Dominant frequency of IEPE sensor channel 1	mHz
P6	Dominant frequency of IEPE sensor channel 2	mHz
P7	Supply voltage of IEPE sensor channel 1	mV
P8	Supply voltage of IEPE sensor channel 2	mV
P9	Analogue supply voltage (~19 V)	mV
P65	Frequency of digital input 1	mHz
P66	Frequency of digital input 2	mHz

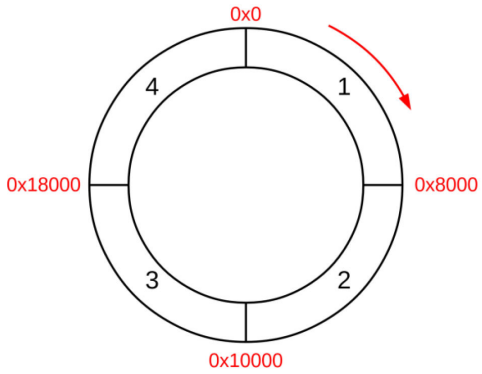
The individual filters can be disabled via parameters 11 and 21:

Bit	Function
0	Channel on/off
1	High pass on/off
2	Velocity (integrator on), Acceleration (integrator off)
3	Low pass on/off
4	Biquad filter on/off

VORLÄUFIGE VERSION / PRELIMINARY VERSION

2.3. Signal storage

The sampled sensor signals can be stored directly in an external RAM of the microcontroller. 131072 (0x20000) samples can be stored per channel. The memory is divided into four segments:



Sampling memory

The memory is activated via parameter 50. Both permanent sampling of the sensor signals and event-controlled sampling (trigger function) are possible. If the memory overflows (0x1FFFF), storage starts again at address 0.

Parameter 50 has the following functions:

Bit	Function
0	Sampling off
1	Permanent
2	Triggered by RMS value of channel 1
3	Triggered by RMS value of channel 2
4	Triggered by RMS value of channel 1 | channel 2 (OR-linked)
10	Triggered by digital input 1 (L/H transition)
11	Triggered by digital input 1 (H/L transition)
12	Triggered by digital input 2 (L/H transition)
13	Triggered by digital input 2 (H/L transition)
14	Triggered by frequency of digital input 1 (rising)
15	Triggered by frequency of digital input 1 (falling)
16	Triggered by frequency of digital input 2 (rising)
17	Triggered by frequency of digital input 2 (falling)

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Bit	Function
>21	Reset write pointer, disable sampling, disable trigger.

After the trigger event, parameter 50 is automatically set to 0.

The trigger threshold for trigger sources 2–4 and 14–17 can be adjusted using parameters 51 and 52.

The trigger point can be moved within the sampling window using a pretrigger function. If parameter 53 = 0, the pretrigger is disabled. After the trigger event, exactly 131072 (0x20000) samples are recorded.

To use the pretrigger, first write the desired pretrigger value in parameter 53, e.g. 0x8000.

Then activate permanent sampling (P50 = 1). With this, the memory is filled with current values. The actual trigger is activated by configuring parameter 50 with the respective trigger source.

In this case, exactly 98304 samples (0x20000 – 0x8000 = 0x18000) are recorded after the trigger event. Before the trigger is activated, you must ensure that the pretrigger memory is filled. This can be done via bit 4 of parameter 10.

To subsequently read the samples from the memory, the start of the sample window can be determined as follows: current address pointer (P61, P62) + 1.

2.4. Creating a frequency spectrum using FFT

A frequency spectrum can be generated from the data in the RAM using Fast Fourier Transform (FFT). The length of the FFT corresponds to 2^15 (= 32768). This is a quarter of the sample window. The sampling memory of each channel is divided into four segments. The segment from which the FFT is executed depends on the current address pointer (parameters 61 and 62). If, for example, the address pointer is at position 0x8001 when the FFT is started, the FFT is executed with the data from segment 1 (Figure “Sampling memory”). This means that the last fully completed segment is always used. This makes it possible to perform an FFT in parallel with sampling.

The FFT function is controlled via parameters 55 and 56:

- An FFT calculation can be triggered via parameter 55.
- Parameter 56 is used to configure the FFT result and to switch between linear and logarithmic display (in dB).

Param-eter	Function
P55	FFT control 0: No calculation 1: Start FFT calculation of channel 1 2: Start FFT calculation of channel 2 Parameter is deleted after FFT calculation

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Param-eter	Function
P56	Reference level in mV for FFT calculation in dB If 0: FFT result absolute with 0.1 µV resolution

2.5. Reading out the data via Modbus

ADC sample data and FFT data can be read via Modbus using the function code “Read input registers (0x04)”.

Depending on the selected address range (parameter 60), access to the respective data is possible:

Parameter 60	Data array	Modbus address range
0x03–0x0A	Storage of channel 1 (8 memory areas of 64 kilobytes each)	0x0–0x7FFF
0x0B–0x12	Storage of channel 2 (8 memory areas of 64 kilobytes each)	0x0–0x7FFF
0x13	FFT of data channel 1	0x0–0x7FFF
0x14	FFT of data channel 2	0x0–0x7FFF

The current sample pointer can be read via parameter P61 (channel 1) or P62 (channel 2). The pointer cannot be modified via the Modbus interface.

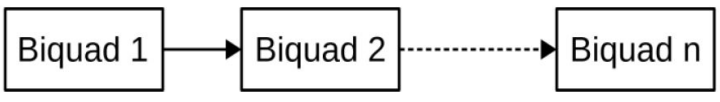
The data is stored as 32-bit signed integers.

2.6. Configuration of an IIR filter

Each channel has a configurable IIR filter. This filter consists of several biquad elements with the following mathematical representation:

$$H(z) = \frac{b_2 z^{-2} + b_1 z^{-1} + b_0}{a_2 z^{-2} + a_1 z^{-1} + a_0}$$

Several of these filter elements can be calculated one after the other in real time. However, the number is limited by the computing power of the microcontroller.



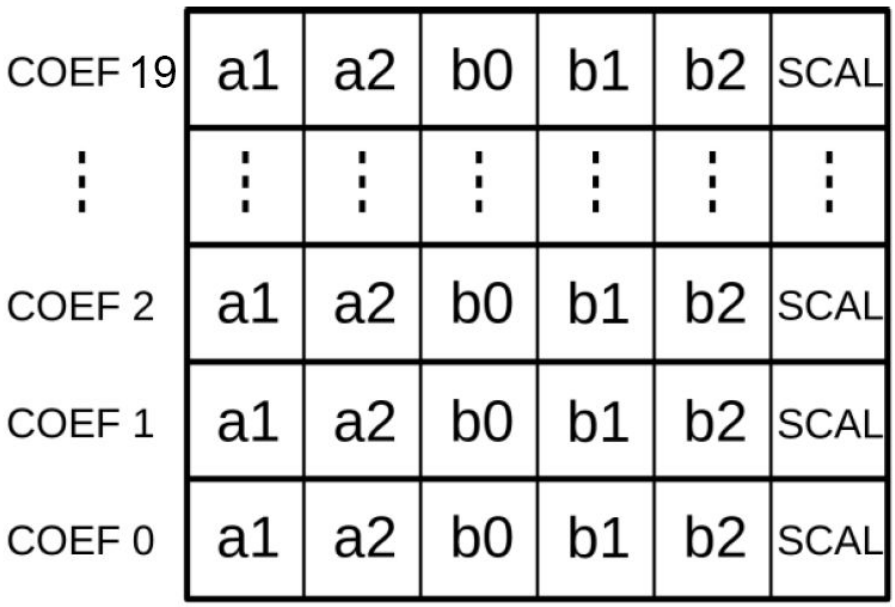
IIR filter

The IIR filter coefficients are managed in an internal coefficient memory.

The coefficient a_0 is always one.

VORLÄUFIGE VERSION / PRELIMINARY VERSION

A maximum of 20 biquad coefficients can be saved. The coefficients are stored as 32-bit signed integers. On top of this, there is a scaling value between 0 and 31. This can be used to determine the bitwise shift of the coefficients (typically between 16 and 31).



IIR-coefficient memory

The coefficient memory can be described using parameters 70–76. The coefficients can also be stored in the internal EEPROM.

First transfer the coefficient data using parameters 71–76. You can then use parameter 70 to store the coefficients at the corresponding location in the memory.

If the “Read” function is executed, the saved coefficients are available in parameters 71–76.

If the coefficient memory is transferred to the EEPROM, it is available again after resetting or after the power supply is interrupted.

Param-eter	Function
P70	Bits 0–15: Coefficient number (0–19) Bits 16–17: Function 1: Read 2: Write 3: Save all biquad coefficients (0–19) to EEPROM
P71	Coefficient a_1
P72	Coefficient a_2
P73	Coefficient b_0
P74	Coefficient b_1

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Parameter	Function
P75	Coefficient b_2
P76	Scaling between 0 and 31 (bitwise shift of the coefficients)

The actual IIR filter can be configured using parameters 17, 18 and 27, 28:

Parameter	Channel 1	Channel 2	Comment
First biquad coefficient (coefficient memory)	P17	P27	Between 0 and 19
Number of biquad elements	P18	P28	P17 + P18 ≤20 P27 + P28 ≤20
Activate IIR filter	P11	P21	Bit 4

For example: P17 = 0, P18 = 4

The IIR filter of channel 1 consists of 4 biquad elements. The corresponding coefficients are loaded from the coefficient memory from position 0.

To activate the IIR filter of the respective channel, set bit 4 of parameter P11 or P21. Please note, however: When changes are made, all bits are written in the register.

Parameter 82 (channel 1) and 82 can be used to check the computing power. Since the IIR filter is executed 48000 times per second (after each sampling), monitoring of the computing time is necessary.

Parameters 81 and 82 should not be greater than 40.

VORLÄUFIGE VERSION / PRELIMINARY VERSION

3. Configuring parameters via the MQTT protocol

MQTT protocols can be used to publish parameters and data arrays (ADC and FFT data). Parameter configuration is also possible via MQTT. The Ethernet interface is configured via parameters 91–94.

Parameter	Function	Default setting
P91	IP address	192.168.1.200
P92	Net mask	255.255.255.0
P93	Gateway address	192.168.1.1
P94	MQTT server address	192.168.1.2

Port: 1883

After a change, the parameters must be saved to EEPROM. The transfer takes place after the reset:

P100 = 0 (disable device)

P102 = 1 (save parameters to EEPROM)

P102 = 0xffff0000 (reset device)

Configuration/task		
Change parameter	Topic	vm102/dev<device serial number>/paramset/
	Message	Parameter values in ASCII (32-bit signed integers)
	Example	Topic: vm102/dev00142DE082EB/paramset/p12 Message: 1000
Publish parameter	Topic	vm102/dev<device serial number>/parameter/
	Message	Parameter values in ASCII (32-bit signed integers)
	Example	Topic: vm102/dev00142DE082EB/parameter/p12 Message: 1000
Publishing of parameters can be forced via the following topic	Topic	vm102/dev<device serial number>/parampublish/
	Message	Parameter values in ASCII: “1,2,10,11” or “1–11” (32-bit signed integer)
	Example	Topic: vm102/dev00142DE082EB/parampublish/p12 Message: 1,2,10,11
Autopublish for parameters 1–9	Parameters are published automatically at regular intervals Parameter 78: Interval in seconds	

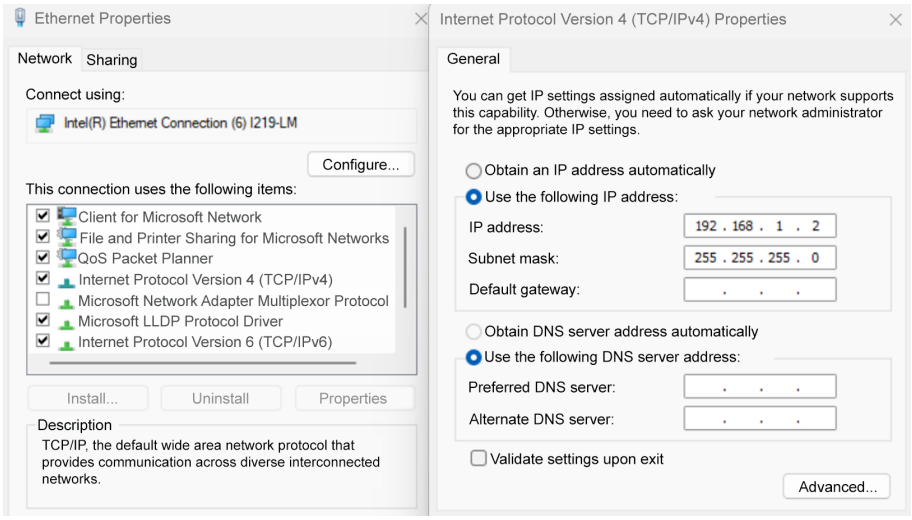
VORLÄUFIGE VERSION / PRELIMINARY VERSION

Configuration/task		
	Parameter 79: activate corresponding parameter (Bit1 > P1, Bit2 > P2, etc.)	
Publish ADC raw data and FFT data	Topic	Vm102/dev<device serial number>/datapublish/ch1 Vm102/dev<device serial number>/datapublish/ch2 Vm102/dev<device serial number>/datapublish/fft1
	Message	“start=<start address of storage range starting with 0>,len=<number of data, max. 100>,topic=<topic with which the data can be published, max. 50 characters>”
	Example	Topic: vm102/dev00142DE082EB/datapublish/ch1 Message: start=0,len=20,topic=vm102/dev00142DE082EB/rawdata/sector0
Error, e.g. incorrect parameter or address, message length exceeded, etc.	Topic	vm102/dev<device serial number>/status
	Message	–
Heartbeat	Topic	vm102/dev<device serial number>/hb
	Message	“vm102” + counter value
	Example	Topic: vm102/dev00142DE082EB/hb Message: vm102 hb: 34

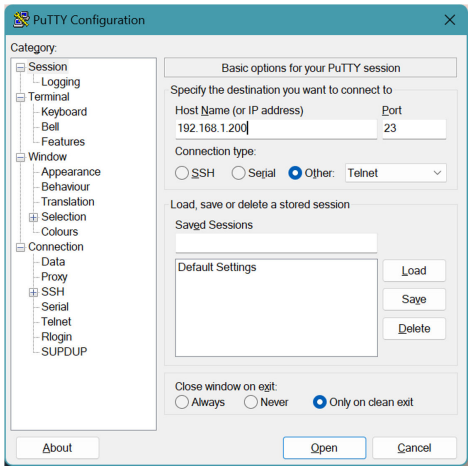
VORLÄUFIGE VERSION / PRELIMINARY VERSION

4. Changing the IP address of the HUB-VM102

1. Connect the Ethernet port of the HUB-VM102 to the Ethernet port of your PC.
2. To communicate with the module, change the IP settings of your PC's network connection as follows:



3. Install the **PuTTY** programme and open it.
4. In the **PuTTY configuration** dialogue window, enter the new IP address under **Host name (or IP address)** and select the **Other** option under **Connection type**.



Example of the PuTTY dialogue window

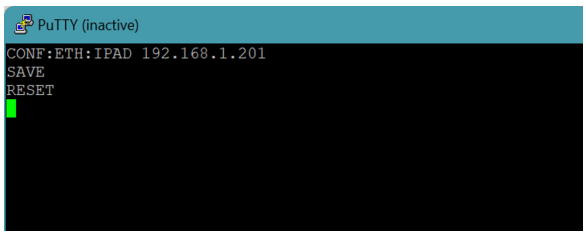
5. Click **Open** to access the PuTTY console.
6. Now enter the following three commands one after the other and confirm each entry with Enter:

   a. `CONF:ETH:IPAD xxx.xxx.xxx.xxx`
   xxx.xxx.xxx.xxx = your new IP address

   b. SAVE

   c. RESET

VORLÄUFIGE VERSION / PRELIMINARY VERSION



7. Check the newly configured IP address. On the PuTTY console, enter the command `CONF:ETH:IPAD?`

   The new IP address should then be displayed.

VORLÄUFIGE VERSION / PRELIMINARY VERSION

5. Parameter directory

5.1. Parameters for signal processing

Parameters, channel 1	Parameters, channel 1	Description
P1	P2	RMS value in µV
P3	P4	Peak value in µV
P5	P6	Dominant frequency of IEPE sensor in mHz
P7	P8	Supply voltage of IEPE sensor in mV
P9 (channel independent)		Analogue supply voltage (~19 V)
P10 (channel independent)		Status: Bits 0–1: Sampling status, channel 1 Bits 2–3: Sampling status, channel 2 0 = No sampling 1 = Permanent sampling, no trigger activated or disabled 2 = Sampling, trigger has been activated Bit 4: Pretrigger memory is filled
P11	P21	Channel configuration: Bit 0: Channel ON/OFF Bit 1: High pass on/off Bit 2: Velocity (integrator on), Acceleration (integrator off) Bit 3: Low pass on/off Bit 4: Biquad filter on/off
P12	P22	Gain correction in ‰ steps (default = 1000)
P13	P23	High-pass filter threshold frequency in mHz
P14	P24	Integrator threshold frequency in mHz
P15	P25	Low-pass threshold frequency in Hz
P16	P26	RMS value threshold frequency in mHz
P17	P27	First biquad coefficient
P18	P28	Number of biquad elements
P31	P41	Peak detector rise time (µs)

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Parameters, channel 1	Parameters, channel 1	Description
P32	P42	Peak detector fall time (ms)
P33	P43	Minimum signal size for peak detection (P5, P6) in mV
P50 (channel independent)		Sampling configuration: 0: Sampling OFF 1: permanent 2: triggered by RMS value of channel 1 3: triggered by RMS value of channel 2 4: triggered by RMS value of channel 1 | channel 2 10: triggered by digital input 1 (L/H transition) 11: triggered by digital input 1 (H/L transition) 12: triggered by digital input 2 (L/H transition) 13: triggered by digital input 2 (H/L transition) 14: triggered by frequency of digital input 1 (rising) 15: triggered by frequency of digital input 1 (falling) 16: triggered by frequency of digital input 2 (rising) 17: triggered by frequency of digital input 2 (falling) >21: reset write pointer, switch off sampling, finish trigger-sampling After the trigger event, parameter P50 is automatically set to 0.
P51	P52	Trigger threshold in µV (P50 = 2–7) or mHz (P50 = 14–17)
P53 (channel independent)		Pretrigger (in samples 1/48000 Hz)
P55 (channel independent)		FFT control 0: No calculation 1: Start FFT calculation of channel 1 2: Start FFT calculation of channel 2 Parameter is deleted after FFT calculation
P56 (channel independent)		Reference level in mV for FFT calculation in dB If zero: FFT result absolute with 0.1 µV resolution
P60 (channel independent)		Address register, data exchange 0x0: Device data Serial no.: 4 byte Firmware rev.: 4 byte

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Parameters, channel 1	Parameters, channel 1	Description
		Hardware rev.: 4 byte Device name: 16 byte (“HUB-VM102”) 0x01: RMS buffer, channel 1 (4096 bytes) 0x02: RMS buffer, channel 2 (4096 bytes) 0x03–0x0A: Storage, channel 1 (8 memory areas of 64 kB each) 0x0B–0x12: Storage, channel 2 (8 memory areas of 64 kB each) (ADC raw values) 0x13: FFT, data channel 1 (64 kB) 0x14: FFT, data channel 2 (64 kB) (if P56 = 0: linear with 0.1 µV resolution; otherwise in 0.001 dB relative to P56) 0x0808 0000: unused flash bank for firmware update
P61	P62	Current write pointer of SDRAM data array
P63	P64	Current write pointer of RMS data array
P65	P66	Frequency of digital input in mHz

5.2. Parameters for device configuration

Parameter designation	Description	Parameter
Update biquad coefficient	Bits 0–15: Number (0–39) Bits 16–17: 1: Read 2: Write 3: Save all biquad coefficients (0–39) to EEPROM	P70
	Coefficient a_1	P71
	Coefficient a_2	P72
	Coefficient b_0	P73
	Coefficient b_1	P74
	Coefficient b_2	P75
	Scaling between 0 and 31 (bitwise shift of the coefficients)	P76

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Parameter designation	Description	Parameter
DSP computer utilisation, channel 1	Should be between 25 and 40 (%)	P81
DSP computer utilisation, channel 2	Should be between 25 and 40 (%)	P82
MQTT autopublish period	in seconds	P88
MQTT autopublish enable parameters 1–9	Bit 1: Parameter 1 Bit 2: Parameter 2 ...	P89
IP address	192.168.1.200	P91
Net mask	255.255.255.0	P92
Gateway address	192.168.1.1	P93
MQTT server address	192.168.1.2	P94
Switching threshold, dig. input	Threshold in mV	P95
Modbus ID	1–254 (then save parameters in EEPROM + reset) If the parameter is outside the permissible range, the default value (1) is used	P96
Modbus baud rate	9600–115200–1000000 (then save parameters in EEPROM + reset) If the parameter is outside the permissible range, the default value (115200) is used	P97
Sampling frequency	48000, 12000 (4× oversampling), 6000 (8× oversampling) (then save parameters in EEPROM + reset) If the parameter is outside the permissible range, the default value (48000) is used	P98
Device control reg.	0: OFF 1: Normal operation	P100
Device error / status reg.	Bits 0–15: Status Bit 15: next Modbus device is enabled Bits 16–31: Error Bit 16: Watchdog reset	P101

VORLÄUFIGE VERSION / PRELIMINARY VERSION

Parameter designation	Description	Parameter
	Bit 17: EEPROM error Bit 18: Parameter error (invalid value range)	
Device config.	Bit 0: Store parameter to EEPROM (only in device mode OFF) Bits 16–31: 0xffff: Device reset (only in device mode OFF)	P102
Backplane bus Enable neighbouring module	0: Module disabled (default) 1: Module released	P103

VORLÄUFIGE VERSION / PRELIMINARY VERSION

This document is available in electronic form in the download portal of in.hub. Printed versions or copies not explicitly provided by in.hub are deemed uncontrolled.

The original language of this document is German.

Made in Germany.

Service & Support: service@inhub.de | https://community.inhub.de/

in.hub Download portal: https://download.inhub.de/

in.hub GmbH
Technologie-Campus 1
DE-09126 Chemnitz

+49 371 335 655 00
info@inhub.de

VORLÄUFIGE VERSION / PRELIMINARY VERSION